

Guía de referencia rápida para dominar

JAVA WORKSHOP

pcmanía

15 años
HOBBY PRESS

Edita: Hobby Press S.A.
Redacción: Ernesto Martí, Mónica Ballartín.
Edición y Diseño: Equipo PCmanía.

Imprime: Pentacrom S.L.
Depósito legal M-34844-92
Este suplemento se incluye
conjunta e inseparablemente con PCmanía



Guía de referencia rápida para dominar

JAVA WORKSHOP

pcmanía

Presentación

Este libro es un regalo especial de PCmanía. En su interior desvelamos todos los secretos del lenguaje con más futuro en la red, el Java, estudiando a fondo «Java WorkShop», un programa desarrollado por Sunsoft que hará las delicias de todos los amantes de Internet. En el CD-Rom de portada incluimos, en rigurosa exclusiva para los lectores de Pcmanía, una versión Beta del programa con la que podéis seguir nuestras explicaciones y participar, además, en un interesante concurso de programación que pone a prueba vuestra imaginación.

«Java WorkShop» permite la creación de "Applets", aplicaciones desarrolladas en Java que, entre otras cosas, permiten incorporar sonidos o música de fondo en las páginas web, crear juegos on-line, incorporar animaciones en las páginas html, realizar menús desplegables en los gráficos de nuestros webs, cambiar el color de fondo de las páginas, etc. Tendréis el placer de experimentar con un lenguaje muy potente que os permitirá dar más vida a vuestras propias páginas webs.

Toda la redacción de PCmanía confía que el notable esfuerzo que ha significado tener la oportunidad de ofreceros este libro, reciba una aceptación unánime entre todos vosotros. Esa ha sido nuestra principal pretensión. ¡Qué lo disfrutéis!

Índice

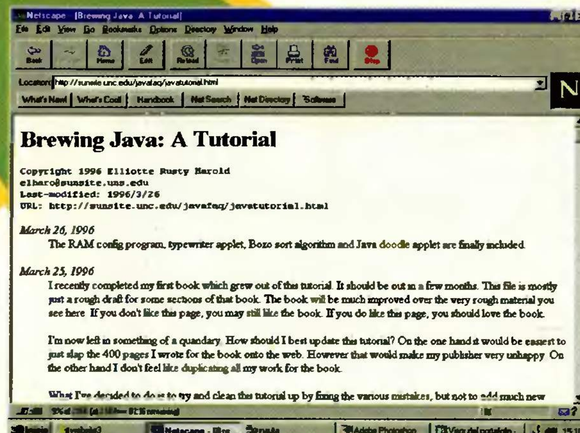
EL LENGUAJE JAVA

Capítulo 1

La repercusión que está teniendo el lenguaje Java no tiene precedentes. Sus características hacen de él una herramienta indispensable para todos aquellos que estén interesados en el desarrollo de páginas en Internet, o apuntarse al carro de los nuevos lenguajes de programación.

El proceso de creación del lenguaje ha sido largo y laborioso, puesto que su desarrollo comenzó hace bastante tiempo viéndose interrumpido, entre otras cosas, por los cambios a nivel empresarial de sus promotores. El 23 de mayo de 1995, Sun Microsystems anunció for-



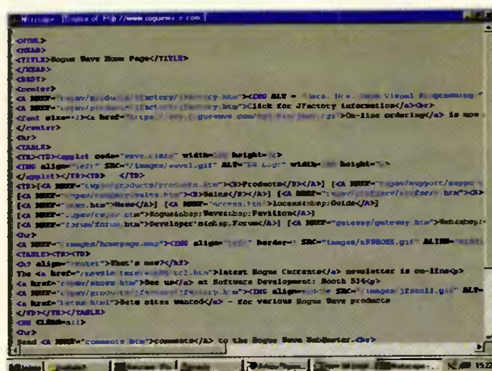


malmente la aparición del nuevo lenguaje de programación Java y HotJava, un navegador para éste desarrollado con el mismo lenguaje.

El objetivo principal de Java es aunar los procesos necesarios para reunir, en el contexto del desarrollo de software, los distintos entornos de sistemas operativos y hardware. También se requería un lenguaje de programación que permitiera la transferencia de datos entre distintas plataformas con seguridad y con un mínimo consumo de recursos del sistema.

En un principio, Java formaba parte de un proyecto para encontrar un software válido para la amplia variedad de sistemas operativos que existían en los ordenadores de los usuarios de informática. En conjunto, el objetivo era conseguir un lenguaje que operara en tiempo real, que fuera multiplataforma, y no exigiera un exceso de recursos.

Cuando se comenzó a trabajar en el proyecto se pensó en desarrollarlo a partir del lenguaje C++, sin embargo, se debían solventar los errores que se registra-



ban y eliminar algunas dificultades de programación. Por eso al final se optó por un lenguaje nuevo que aunara las ventajas de los ya existentes pero evitara los inconvenientes. Como resultado apareció el lenguaje JAVA, conocido en los círculos profesionales como OAK.

Algunos usuarios se pueden haber sorprendido cuando visionando un web en Internet hemos visto en la barra de estado la siguiente observación: "Running applet...."; un applet, como puede que ya hayáis descubierto, es la aplicación que se ha realizado mediante el lenguaje Java y que después se ha incorporado en el código HTML de la página web.

Otro de los conceptos utilizados por Sun Microsystems en sus presentaciones es el de "multithreading", esto quiere decir que un usuario que está creando aplicaciones puede hacerlo con varias peticiones de programa distintas, lo que le da mayor interactividad a las aplicaciones, consiguiéndolo gracias a la suma de varias primitivas.

En el mismo sentido Java ofrece seguridad en sus "Thread", es decir en las peticiones que hace el programa, además la funcionalidad de las librerías hace que las posibilidades de conflicto sean mínimas.

Java corrige y mejora las capacidades de los lenguajes en que se basó, integrando en sus especificaciones los protocolos de transmisión TCP/IP, HTTP... Uno de sus logros es, también, haber conseguido compilar los códigos fuente en códigos de bytes facilitando la transmisión entre distintas plataformas.

Sin embargo y, a pesar de lo que pueda parecer este lenguaje, no se originó exclusivamente para Internet, sino que con él se pueden desarrollar las mismas aplicaciones que con, por ejemplo, el lenguaje C, C++..., eso sí con mayor seguridad en sus cálculos, puesto que no utiliza punteros aritméticos, ofrece también más seguridad de cara a los crackers...

El lenguaje Java no es especialmente difícil, pero para trabajar cómodamente requiere unos ciertos conocimientos de programación de los lenguajes C y C++, entre otros. Debido a la necesidad de ampliar su utilización, y no quedar reducido al ámbito de los desarrolladores profesionales o a aquellas personas con un buen nivel de programación, se han contemplado programas de creación de applets que no presentan gran dificultad.

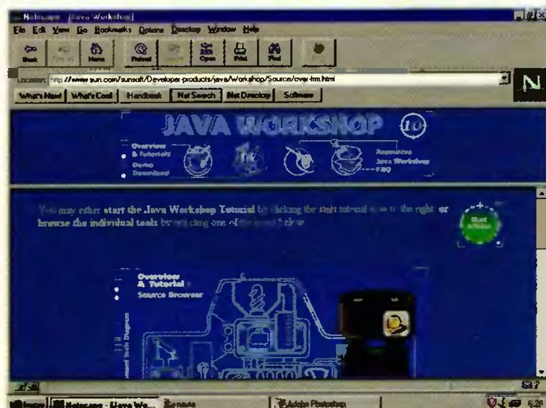
Mediante este programa podemos crear páginas web con applets interactivos sin tener que teclear más que un mínimo de código (o nada), es decir prácticamente sin saber programar.



JAVA WORKSHOP 1.0

Capítulo2



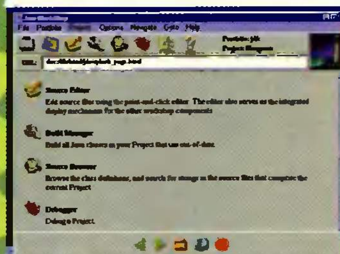


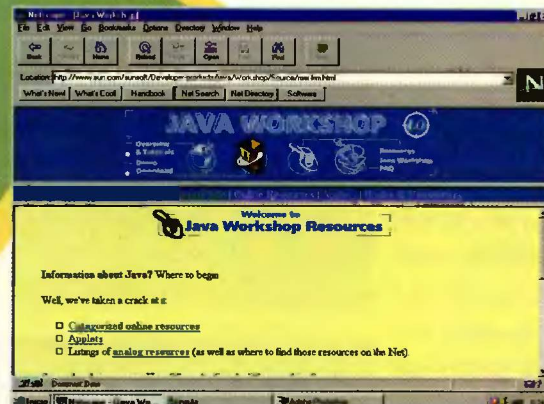
Java Workshop 1.0 es una versión beta del programa que verá la luz en breve. Este presenta una serie de características que lo hacen realmente útil y cómodo, entre ellas podemos comentar las siguientes:

• Herramientas de desarrollo gráficas de fácil utilización, permitiendo creación de webs sin dificultad.

• El conjunto de estas opciones está completamente integrado, evitando tiempos muertos al tener que pasar información de una a otra. De este modo se consigue agilizar la creación.

• Sus herramientas nos permiten ordenar nuestros proyectos en carpetas y tener siempre presente los elementos que constituyen cada una de las aplicaciones con sus características.





• Podemos editar el código permitiendo que se consulte desde la red, y al mismo tiempo pudiendo consultar los códigos de otras aplicaciones existentes.

• Su capacidad de multiplataforma, que permite que las aplicaciones y applets funcionen correctamente y mantengan su estética, sea cual sea la plataforma sobre la que "corran".

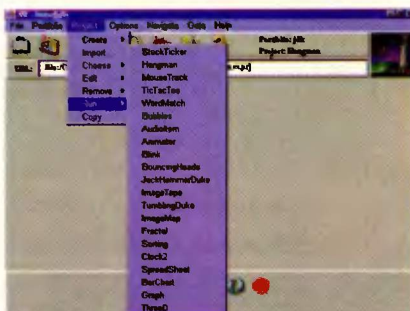




• Su sistema de compilación permite la localización de errores y su resolución de una manera rápida y eficaz.

En resumen, ofrece la posibilidad de gestionar y crear de una ma-







nera cómoda aplicaciones y applets con lenguaje Java, aunque habrá que esperar a la versión definitiva para comprobar sus capacidades reales, puesto que ya habréis notado que, como suele ocurrir con las versiones beta, no ofrece todas las opciones a pleno rendimiento.

El entorno de desarrollo de Java Workshop se basa en un "web browser" que permite trabajar a partir de una sencilla barra de herramientas. Ésta tiene el mismo funcionamiento que las utilizadas en programas similares, permitiéndonos, entre otras cosas, cambiar de herramienta simplemente yendo hacia delante o hacia atrás del mismo modo que si nos encontráramos navegando.

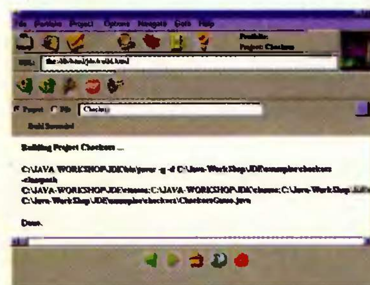
Novedades previstas en futuras versiones de Java WorkShop

En un futuro próximo, está previsto que las versiones posteriores de «Java WorkShop» incluyan una serie de herramientas que facilitarán enormemente la creación, no sólo de applets, sino tambien de páginas web de una forma gráfica e intuitiva. Estas son las mejoras que, en principio, están previstas:

• Un interface gráfico para desarrollar fácilmente aplicaciones Java.

• Un editor "WYSIWYG" (lo que ves es lo que obtienes) de HTML, que permitirá un gran control en el diseño y el control de nuestras páginas web.

• Una versión para ordenadores Mac.





• Un "Just-In-Time" (JIT). Este es un compilador especial que, convirtiendo los códigos de "byte" generados por Java, en código nativo de máquina, aumenta la velocidad de las aplicaciones.

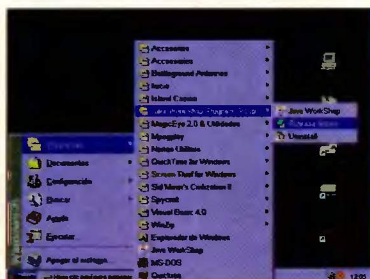
Está previsto que la versión final de Java-WorkShop esté definitivamente terminada en mayo de este año, y estará disponible en CD-ROM.

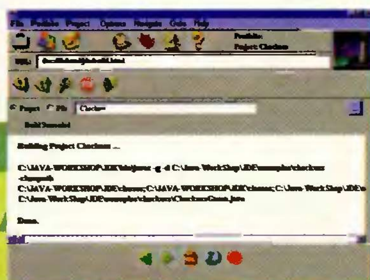
REQUISITOS E INSTALACIÓN

Capítulo3

La verdad es que los requisitos que vamos a necesitar para instalar el programa son bastante elevados, lo cual es lógico si tenemos en cuenta sus capacidades. Además, como ya hemos comentado, en el CD de portada incluimos una versión beta, no os extrañe si, por ejemplo, no redibuja correctamente la pantalla al ir seleccionando las diferentes opciones.







Dos truquillos: primero, si os pasa esto, minimizad la pantalla y volvedla a maximizar ¿se ha solucionado el problema?, si no es así, pulsad el botón de "reload", una o varias veces. A nosotros estas opciones nos han funcionado, y SunSoft nos ha asegurado que estos detalles estarán resueltos en las versiones posteriores. ¡Es el precio de la primicia!

Pasemos ahora a describir los requisitos mínimos para el funcionamiento del programa:

Sistema operativo:

- Windows 95 o Windows NT 3.5.1

Requisitos mínimos de Hardware:

- PC compatible con un procesador Pentium a 90 mhz.
- 16 Mb. de memoria RAM.
- 45 Mb. de espacio libre en el disco duro.
- Unidad de CD-ROM.
- Monitor que acepte una resolución de 1024x768

Instalando el programa...

Antes de instalar el programa deberemos asegurarnos: primero, que nuestro ordenador cumpla las características mínimas requeridas, y segundo, no tener instalada alguna versión del paquete de desarrollo de lenguaje Java llamado JDK, si es así, adelante... (en el supuesto que tengamos instalada una versión del JDK, deberemos borrarla completamente del disco).

Otro punto importante a tener en cuenta en el momento de instalar el software, será cuando nos solicite el nombre del directorio en el que queremos instalar el programa; deberemos tener presente que no han de existir espacios en blanco, si ponemos alguno, no funcionará correctamente. Tampoco funcionará si lo instalamos en la carpeta de windows 95 "program files".

Una vez instalado todo correctamente en vuestro disco duro, veamos un ejemplo de como se trabaja con este programa.

PRIMER CONTACTO
Capítulo4

A continuación os presentamos una descripción de las opciones básicas que ofrece la barra de tareas del programa:

Portfolio Manager

Nos permite la creación de las carpetas para nuestros proyectos de lenguaje Java. Bastante sencillo de utilizar, nos facilita controlar y adaptar objetos y applets a partir de los cuales podremos crear nuevas aplicaciones. Los movimientos entre carpetas los realizaremos mediante el "drag and drop", es decir arrastrar y soltar, lo que nos facilita enormemente las tareas de modificación. A partir de estas carpetas podremos editar los proyectos que deseemos y hacer pública la información (decidiendo nosotros si será mucha o poca), lo que nos permite esta herramienta es acceder, como contraprestación, a las carpetas editadas en otros lugares de la web.



Project Manager

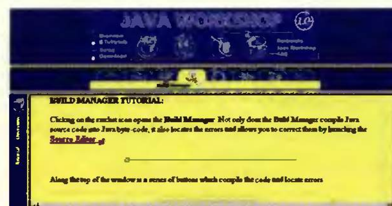
Configura los directorios y preferencias de nuestro proyecto. No tenemos que preocuparnos de memorizar sus características, puesto que esta herramienta se ocupa de ello. Asocia los componentes de sonido, las páginas HTML, los gráficos, los ficheros ejecutables..., es decir todo aquello que constituye el proyecto en sí. También nos ofrece la opción de configurar los parámetros que controlarán algunas de las operaciones a desarrollar durante la creación de la aplicación; por ejemplo, procesos de test, debug,...

Source Editor

Constituye una manera sencilla de crear y editar nuestros códigos fuente, apoyado en el concepto de paquete integrado de este software. Simplemente se trata de hacer click con el ratón en aquel sector que nos interese y nos permite acceder a él para corregir errores o consultarlo. Al mismo tiempo, trabaja con las versiones estándar de control de sistemas: SCCS, RCS, PVCS...

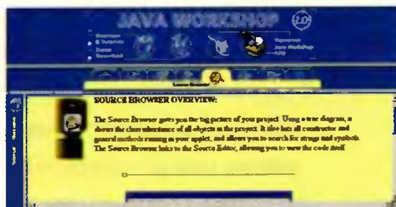
Build Manager

Nos permite compilar el código fuente en código de bytes. Debido a las características del programa, completamente integrado, recoge él mismo la información necesaria para realizar estos procesos; seleccionando únicamente aquellos ficheros que deben modificarse o actualizarse. Por otro lado, nos facilita la localización de errores puesto que si al intentar compilar descubre uno de ellos, abre el editor de fuente y nos enlaza con el error, haciendo mucho más sencilla la corrección.



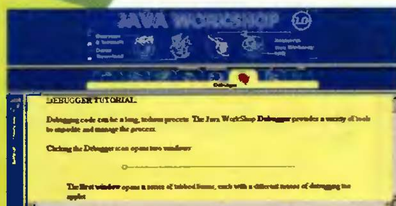
Source Browser

Nos presenta un organigrama de árbol donde se muestran las relaciones de todos los objetos de nuestro proyecto. En este, nos permite ver los ficheros .class del applet y genera una lista en hipertexto que nos enlaza con el código y nos indica los métodos. Por otro lado, nos muestra la conexión entre los objetos que se encuentran en el proyecto y nos permite la búsqueda a partir de símbolos para localizar el código que deseemos.



Debugger

Nos facilita la tarea de buscar los posibles problemas de nuestra aplicación mediante una serie de herramientas de control y dirección del proceso que nos aparecerán en la carpeta-menú. Controla la ejecución de los códigos, de los applets (tanto en nuestro ordenador como en otro remoto) o incluso dentro de una página de HTML o en una aplicación exclusivamente de Java. Pudiendo detener la ejecución en cualquiera de las peticiones del sistema o "threads" en que lo deseemos, nos permite también examinar la petición en ese momento, examinar otras y resumir la ejecución de estas. En caso de que durante la visualización de un applet en el "applet viewer" se encuentre algún problema se arrancará directamente el sistema de control, debido a su condición de paquete integrado.



Applet Tester

Esta parte del programa, también conocida como "applet viewer" o "visualizador de applets" nos permite "correr" nuestros proyectos dentro del entorno del programa y al mismo tiempo testearlos. Podemos visionarlo durante la creación, para ir viendo nuestros progresos, o si lo preferimos una vez finalizado el proyecto. Examina las necesidades que requiere el proyecto para funcionar correctamente. Si lo considera necesario buscará un nuevo navegador para que nuestra aplicación funcione correctamente.



On line Help

Este apartado, que facilita sensiblemente nuestra tarea creativa, se encuentra organizado en seis partes diferentes, accediendo a ellas a partir de la barra de ayuda. Estos son los apartados: contenidos, índice, comenzando, haciendo debugg, creando applets, dirigiéndolos y fuente de navegación.

LA BARRA DE HERRAMIENTAS



En este capítulo vamos a describir las acciones de los iconos que aparecen en la pantalla ordenados de izquierda a derecha.

El icono correspondiente al **Portfolio Manager** nos permite acceder al conjunto de componentes de Java y aplicaciones que guardamos allí. En este lugar podremos guardar nuestros trabajos, ya estén terminados o no. Esta opción comprende también un URL que permite visionarlo mediante un browser compatible con Java, por ejemplo el Netscape 2.0.



Si lo que deseamos es comenzar un nuevo proyecto seleccionaremos la opción de "Create Project", y rellenaremos los distintos campos de información que nos solicitan:

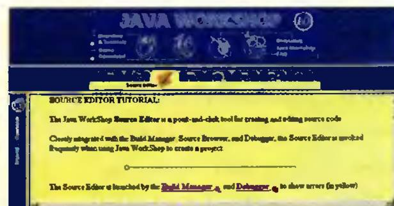
- Nombre: donde incluiremos el que identificará el proyecto.
- Source Directory: introduciremos el nombre del directorio que ha de contenerlo.
- Existing sources: indicaremos si la fuente existe.
- Sources: seleccionaremos la fuente que corresponda.
- Add all in Directory: debemos seleccionarlo para que tenga efecto.
- Apply: si no lo presionamos los cambios no se guardarán.

A continuación nos aparecerá el nombre del proyecto junto con el icono en el programa, con lo que sólo deberemos hacer click en él para iniciar el trabajo con dicho proyecto.

El segundo icono que encontraremos es el del **Project Manager** que utilizaremos para configurar las preferencias de nuestro proyecto mediante las cinco tablas que nos abrirá al seleccionarlo:

- General: selecciona el tipo de proyecto que deseamos crear; un applet que integraremos en nuestras páginas HTML, imágenes, paquetes..., y dónde observaremos que ha tomado los datos de la información del portafolio.
- Build y debug/browse configura los path, los directorios y las preferencias de compilación y debug. Debemos asegurarnos que el nombre de nuestro proyecto está incluido en "Files in:".
- Run y Publish, nos permiten configurar las opciones de test y edición. En la primera de estas opciones deberemos escoger "Main Class File" para seleccionar los path y el nombre de los ficheros. En "Code base" tendremos que escoger el directorio de la fuente. Tan sólo nos faltará ahora seleccionar la anchura y el tamaño.

En **Source Editor**, podemos crear y editar códigos de fuente. Está muy relacionado, debido a sus características de paquete integrado, con las otras herramientas que contienen datos sobre el proyecto desarrollado. Por ejemplo, se comunica directamente con los apartados de Build Manager y Debugger, en cuanto a la corrección de errores.



La siguiente opción es **Build Manager**, que nos llevará a una de las partes más importantes

de la creación con lenguaje Java, esto se debe a que es el momento en que se realizará la conversión de códigos fuente en códigos de bytes y se localizarán los posibles errores.

En la ventana de este apartado encontraremos una serie de opciones para realizar las tareas. Los botones de izquierda a derecha es la siguiente:

• Next error: efectúa la compilación deteniéndose en el próximo error.

• Previous error: permite acceder al error anterior.

• Initiate Build: si una vez que hemos comenzado a compilar se detiene por un error, para volver a iniciar una vez solucionado, oprimiremos este botón.



• Stop: detener la acción.

• Build all: compilar todo el proyecto, permite hacerlo aunque no haya habido variaciones desde la última compilación.

Desde esta opción podemos utilizar el browser, correr un applet,... que viene a demostrar que el programa es un paquete perfectamente integrado.

El siguiente es el botón que corresponde a **Source browser**, que es la opción que nos permite visionar la estructura de árbol del proyecto, permitiéndonos ver al mismo tiempo el orden y las jerarquías.

Una vez que tengamos seleccionado en el portafolio el proyecto que pretendemos visionar, presionaremos el botón correspondiente a este apartado. Una vez hecho esto nos aparecerá una pantalla con la estructura de dicho proyecto, que nos permite también conocer qué sistemas hemos utilizado para construir, para añadir las implementaciones,... Este menú nos permite también realizar búsquedas dentro del código.

Si seleccionamos el icono de **Debugger**, nos encontraremos ante dos ventanas diferentes, la primera de ellas nos ofrece los contenidos:

• Threads/Stack: nos lleva a una lista con los grupos de "threads".

• Expressions: expresiones.

• Breakpoints: nos lista los breakpoints.

• Exceptions: excepciones.

• Classes: las diferentes clases que incluimos en nuestro proyecto.

• Messages: Diferentes mensajes informativos que nos da el programa.

La segunda ventana nos abre el visor de applets, que hace funcionar la aplicación bajo el control del sistema de debug.

En el icono de **Applet Tester**, podremos ver cómo ha quedado nuestra aplicación, comprobando el resultado final como si ya estuviera en Internet y accediéramos libremente a ella.

Este es el paso final en el proceso de creación de javas, es decir, cuando ya hayamos compilado y realizado todas las operaciones necesarias para asegurarnos su correcto funcionamiento.

El último icono nos puede ser útil en nuestro trabajo de desarrollo puesto que es el de ayuda, presentado de forma clara y mediante hipertexto. Aunque hay partes que están en construcción. Pero en la versión definitiva la ayuda será en línea y muy completa.

UN EJEMPLO PRÁCTICO

Capítulo 6



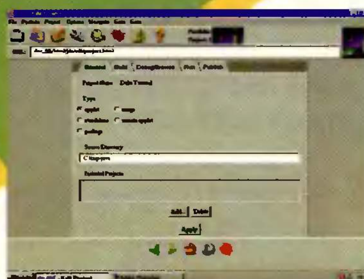


Para que os hagáis una idea de cómo se trabaja con el programa, y ciñéndonos al tópico de que una imagen vale más que mil palabras, a continuación vamos a, utilizando uno de los ejemplos incluidos en el CD-ROM de la revista, ver cómo utilizar el programa para generar applets.

Cuando arrancamos el programa, la primera pantalla que nos aparece, es muy parecida a la de un navegador. En este caso, se trata de una pantalla de introducción al programa y a su funcionamiento, similar al menú ayuda, con enlaces de hipertexto para enlazar los contenidos. Se dispondrá, más adelante, de un servicio on line que en este momento aún no está totalmente desarrollado; recordemos que trabajamos con una versión beta. Para pasar a la pantalla siguiente pulsaremos cualquiera de los iconos.





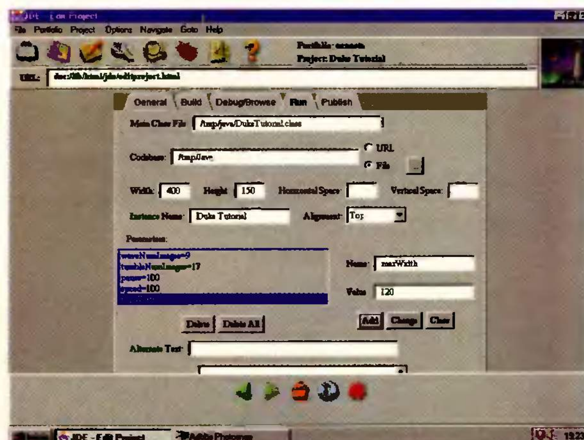


Una vez que ya hemos contactado por primera vez con el programa, y hemos ojeado por encima las herramientas de que dispone, comenzaremos con la creación de nuestro primer applet, siguiendo un sencillo ejemplo.

El primer paso a realizar será crear un nuevo proyecto en el "portfolio manager", para esto seleccionaremos "create a new project" y rellenaremos los distintos campos que aparecen.

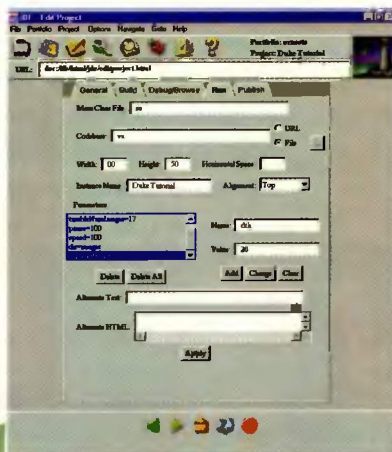
En nombre, introduciremos el nombre de este proyecto-ejemplo: Duke Tutorial , en el directorio de fuente escribiremos: /tmp/java.

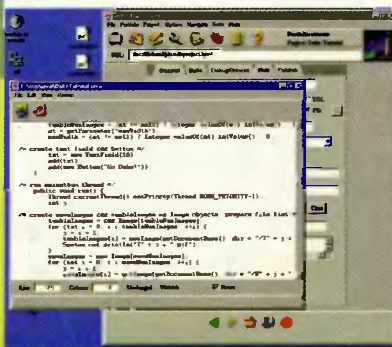
Ahora deberemos especificar si existen fuentes o no, seleccionando en este caso "Yes".



Para rellenar la siguiente casilla ("Sources") presionaremos la opción "Add all in directory", apareciéndonos la fuente Duke Tutorial.java, y cuando ya hemos hecho esto, sólo nos quedará presionar el botón de "Apply" para que tenga efecto.

Si por alguna razón en nuestra toma de contacto ya hubiéramos creado el proyecto ahora deberemos presionar el icono de importar proyecto y seleccionarlo en la ventana que aparezca.

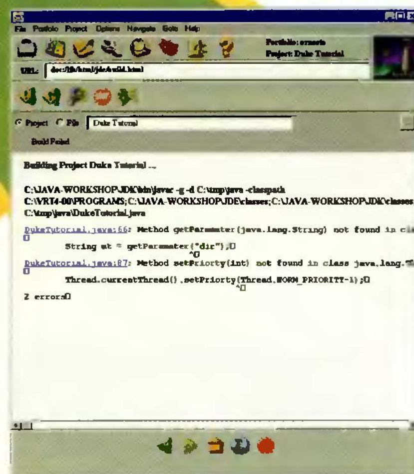




Una vez hecho esto, iremos a la opción de "Project Manager", que aparece con el nombre de "General", en una de las solapas que aparecen en pantalla, allí deberemos rellenar los siguientes pasos:

- Nombre del proyecto: (en nuestro caso) Duke Tutorial.
- En tipo de proyecto deberemos escoger applet.
- En "Source Directory" escribiremos /tmp/java.

Ahora nos dirigiremos a la carpeta de "Run", y la cumplimentaremos:



- En "Main Class File" señalaremos el fichero class que corresponde: /tmp/Java/Duke Tutorial.class
- En URL o File seleccionaremos File.
- En "Codebase" deberemos teclear /tmp/Java
- Las opciones de Tamaño son Width escribiendo 400 y Height con 150.
- En los parámetros, introduciremos los datos en "Name" y en "Value", apareciéndonos después en el cuadro de parámetros. Aunque ahora nos parezca que estamos escribiendo algo sin sentido no es así, se trata del tamaño, la velocidad, las pausas,...

Los valores que deberemos introducir serán estos. El primer valor lo introduciremos en el recuadro llamado "NAME" y el segundo valor en el "VALUE".

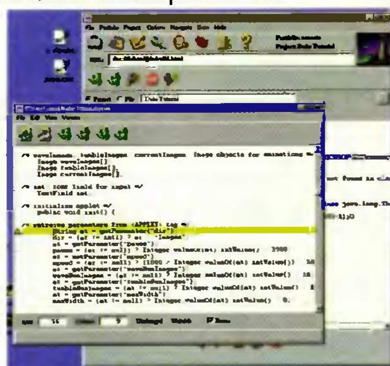
waveNumImages	9
tumbleNumImages	17
pause	100
speed	100
maxWidth	120
dir	images

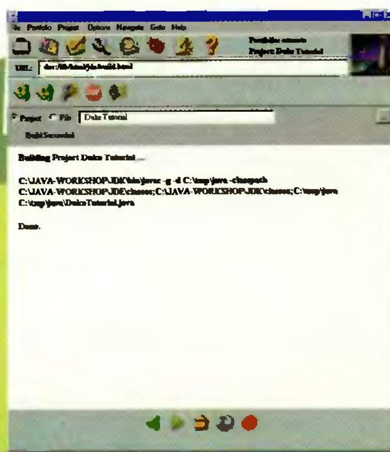
Cada vez que introduzcamos uno de estos valores, con el botón "CLEAR" los limpiaremos para introducir

los siguientes, sabremos que lo hemos hecho bien cuando en el recuadro que está a la derecha aparezcan todos listados.

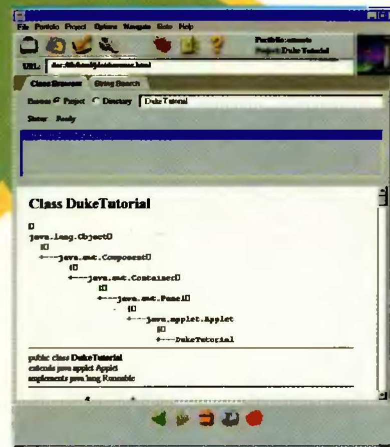
Los botones que encontramos junto al cuadro de parámetros: "Delete" y "Delete all" sirven para eliminarlos, por si nos hemos equivocado al teclear alguno de ellos.

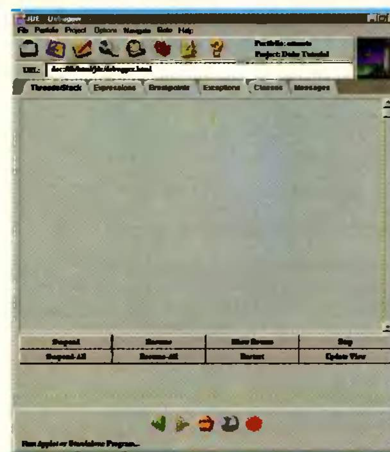
Veremos que debajo de las casillas aparecen unas teclas con varias opciones diferentes. Estas nos permiten realizar algunos procesos imprescindibles. Podemos limpiar los campos en los que introducimos los datos que nos solicita el propio programa; también encontramos la opción de añadir que iremos pulsando para agregar los valores a la casilla de parámetros y, por último, la opción de modificar alguno de los parámetros definidos.





En el "Source Editor", el siguiente paso, aparece una pantalla con dos iconos, el de la izquierda nos permite abrir el código correspondiente a nuestro proyecto. Una vez que lo hemos visionado, deberemos salvarlo con el icono de la derecha.



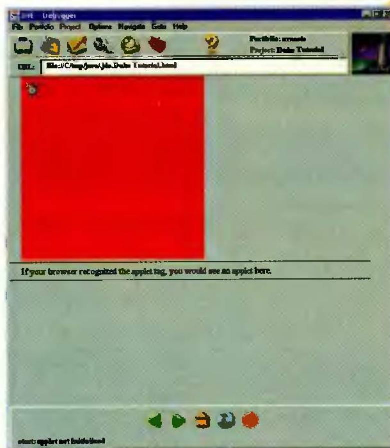


Hecho esto nos dirigiremos al "Build Manager", desde el que compilaremos nuestro proyecto y solucionaremos, si existen, los posibles errores de programación.

Veremos que aparece una pantalla señalándonos que se han detectado dos errores diferentes:

• El primero de ellos aparece en Duke Tutorial.java:66, señalándonos el lugar exacto del error. En este caso nuestro error consiste en que en lugar de Paramater debe poner Parameter.

• El segundo es: Duke Tutorial.java:87, aquí deberemos modificar la palabra Priorty, cambiándola a Priority, su correcta transcripción.



Cuando ya hemos solucionado estos errores le diremos que continúe reconstruyendo en "Iniatiate Build". Al finalizar la construcción nos dirá que se ha ralizado correctamente, "Done".

Para ver el organigrama de los métodos, las interrelaciones, la información de los "threads" del proyecto..., nos dirigiremos al "source browser" dentro del "portfolio manager".

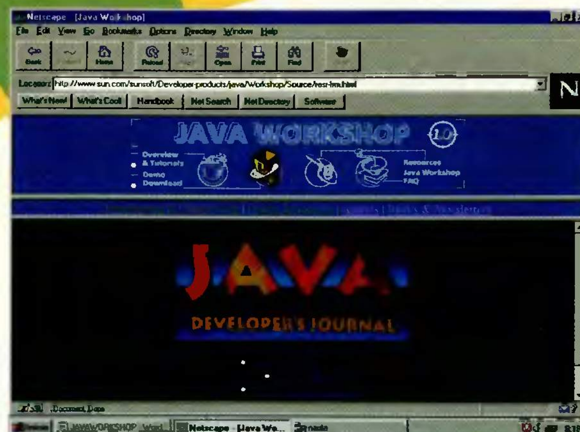
Para hacer el debbug de la aplicación, operación muy tediosa en programación, accederemos al "Debbuger", en la filosofía de concepción del programa, se intentó que las operaciones fueran lo más simples posible, lo han conseguido, este debbuger es realmente sencillo de utilizar. Desde él, podemos lanzar nuestra aplicación para ir controlando los eventos.

La última de la herramientas incluidas en el programa, es el "applet tester" desde la que podemos ver el resultado final de todo nuestro proceso de creación.

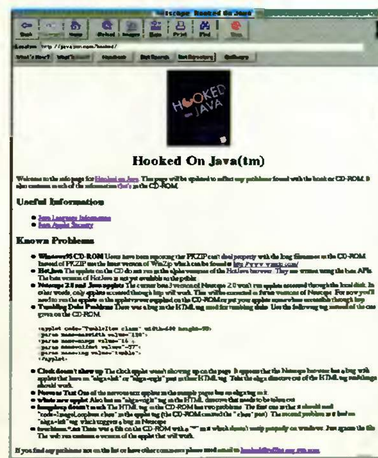
Si tenéis algún tipo de duda o consulta, en la dirección que os hemos dado antes (dirección en Internet para las FAQ del programa) podréis solucionar los problemas que os vayan surgiendo. Animaos a programar en Java, las próximas noticias a cerca de este lenguaje van a ser sin duda espectaculares.

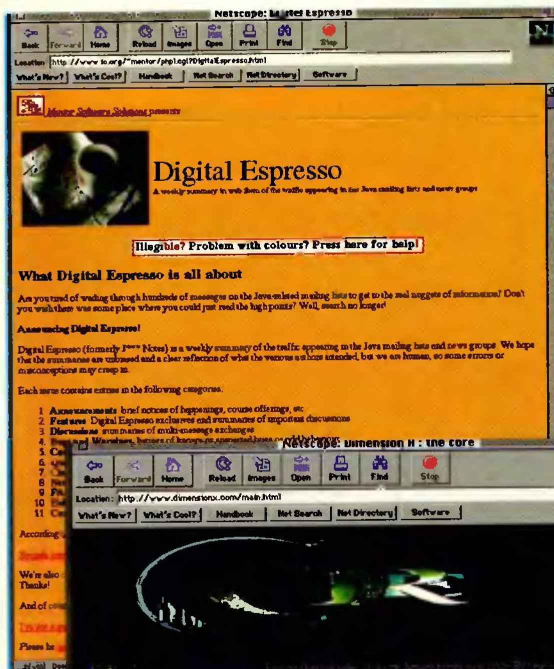
DIRECCIONES DE INTERÉS

Capítulo 7

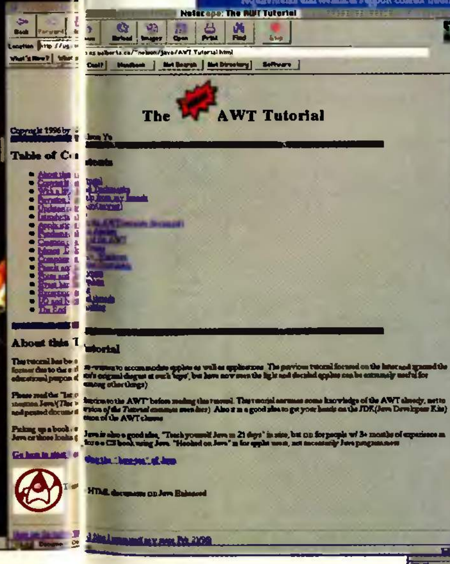


Si queréis más información, acerca del lenguaje Java y del programa Java WorkShop, he aquí un listado de unas direcciones en Internet. En la red podremos encontrar diversos tutoriales, las famosas FAQ (las preguntas más frecuentes acerca de...), ediciones electrónicas de diferentes editoriales y grupos especializados en este entorno de desarrollo, una extensa lista de programas, en definitiva, todo lo necesario para empezar con buen pie en el mundo Java.

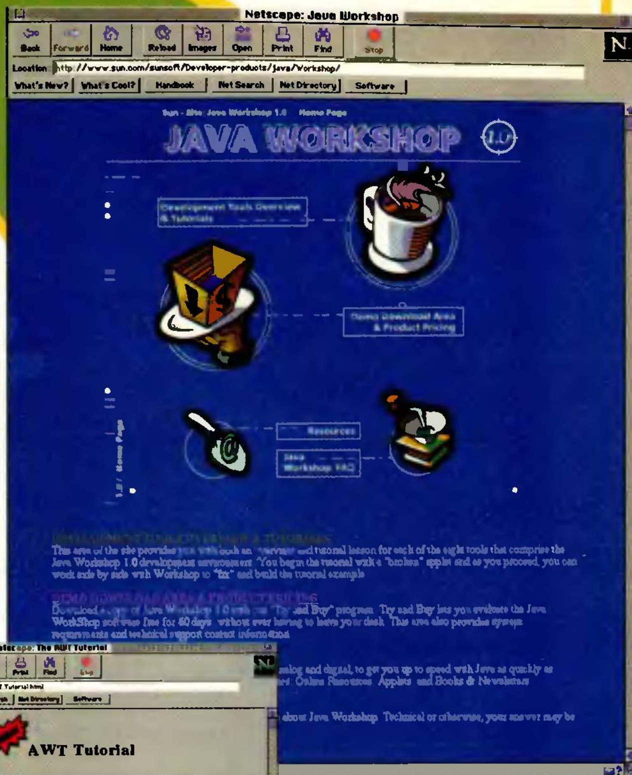




Location: http://www.io.org/~mentor/phpl.cgi?DigitalEspresso.html
Digital Espresso
Illegible? Problem with colours? Press here for help!
What Digital Espresso is all about
Are you tired of wading through hundreds of messages on the Java-related mailing lists to get to the real nuggets of information? Don't you wish there was some place where you could just read the high points? Well, search no longer!
Announcing Digital Espresso!
Each issue contains entries in the following categories:



Netscape: The AWT Tutorial
The AWT Tutorial



Netscape: Java Workshop
Location: http://www.sun.com/sunsoft/Developer-products/java/Workshop/
JAVA WORKSHOP

WORLD WIDE WEB

http://www.javasoft.com
http://www.javasoft.com/hooked
http://www.io.org/~mentor/DigitalEspresso.html
http://www.dimensionx.com
http://ugweb.cs.ualberta.ca/~nelson/java/
AWT.Tutorial.html
http://www.digitalfocus.com/digitalfocus/
faq/howdoi.html
http://amber.wpi.edu/~thethe/Documents/
Besiex/Java/FrIJDE
http://www.gamelan.com

NEWGROUPS:

comp.lang.java
alt.www.hotjava

E-MAIL:

java@java.sun.com

TODO EN UNO:

http://www.sun.com/sunsoft/Developer-products/
java/Workshop/Source/resr-frm.html

En esta dirección, encontraremos un gran número de "links" en los que podremos encontrar muchísima información sobre el lenguaje Java, no sólo tutoriales, también podemos encontrar: mail-lists, grupos de News, publicaciones electrónicas, libros, programas, la lista es extensísima.

Uno de los mejores "links" que podemos hacer desde esta misma dirección, es visitar las página del "Java Developer´s Journal" una publicación electrónica dedicada al lenguaje Java, desde la que podremos saltar a un gran número de lugares relacionados en todos los casos Java.

